JDELI

How to read and write common image file Formats in Java with JDeli

IDR SOLUTIONS

V1.0

April 2023

What is the purpose of this guide?

We get asked a lot of questions about how to read and write different images in Java, So we have written this guide to explain all the image file formats and try to answer your general questions. We also give you code to read and write each format in JDeli.

What is JDeli?

There is already a free image library built into Java called ImageIO. We found it had a number of limitations for our Java software so we wrote our own internal Image Library. It turned out a lot of other Java users also had these problems, so we released JDeli as a separate product for Java Developers.

What problems does JDeli solve?

We wrote JDeli to solve these problems:-

- Pure Java solution to avoid JVM memory crashes due to using native libraries.
- Better Java support for image file formats, especially Tiff, JPEG and JPEG2000
- Java support for Heic images
- Backwards compatibility with existing code through ImageIO plugin and similar API

Can I use JDeli with my existing Java code?

Absolutely! If you are using ImageIO already in your code, the JDeli ImageIO plugin allows that code to use JDeli instead with no changes. Click here for more details.

TABLE OF CONTENTS

BMP files:

What is BMP?

A BMP is a bitmap image file format to store monochrome and colour images.

It was primarily designed for Window platforms although it also works on other platforms. It is not supported in Web browsers.

The file name extension for BMP files is:

```
.bmp
```

How to read a BMP image in Java with JDeli

Step 1: Add JDeli to your class or module path.

Step 2: Create a File, InputStream pointing to the raw BMP image. You can also use a byte[] containing the image data.

Step 3: Read the BMP image into a BufferedImage

```
BufferedImage image = JDeli.read(bmpFile);
```

How to write out an image as a BMP file with JDeli

Step 1: Add JDeli to your class or module path.

Step 2: Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.bmp"));
```

Step 3: Pass image, BMP type and File (or OutputStream) object into the write method

```
JDeli.write(bufferedImage, "BMP", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.BMP, file);
```

or pass in a BmpEncoderOptions object for more control over BMP image output.

```
BmpEncoderOptions options = new BmpEncoderOptions();
JDeli.write(bufferedImage, options, file);
```

DICOM files:

What is DICOM?

DICOM stands for Digital Imaging and Communications in Medicine. DICOM is a standard for medical images. It sets out how to store, exchange and transmit them. You can read more about DICOM images in our article. What is a Dicom Image.

The filename extension for DICOM files is:
`.dcm`

How to read a DICOM image in Java with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File handle, InputStream pointing to the raw DICOM image. You can also use a byte[] containing the image data.
Step 3 Read the DICOM image into a BufferedImage

```
BufferedImage image = JDeli.read(dicomFile);
```

EMF files:

What is an EMF file format?
EMF file format is a public vector image format invented by Microsoft and very popular on the Windows platform.

JDeli can read EMF files directly into Java as a Buffered Image. This can be manipulated in Java and saved to another file format.

The filename extension for EMF files is:

```
 .emf
```

How to read an EMF image in Java with JDeli
Step 1: Add JDeli to your class or module path. (download the trial jar).
Step 2 : Read the EMF image into a BufferedImage

```
//Read Image (can also be OutputStream or byte array)
BufferedImage image = JDeli.read(emfFile);
```

GIF files:

What is a GIF?

GIF stands for Graphics Interchange Format. It is a lossless, bitmap image format which became popular on the world wide web because it supports transparency and simple animations in browsers. It can support up to 256 different colours from a 24bit range of RGB values. It uses LZW compression (which was subject to patents owned by Unisys). Issues over this in 1990s, led it to be largely replaced by the GIF format in modern usage.

The file name extension for GIF files is:

```
.gif
```

How to read a GIF image in Java with JDeli

Step 1 Add JDeli to your class or module path. (download the trial jar).
Step 2 Create a File handle or InputStream pointing to the raw GIF image. You can also use a byte[] containing the image data.
Step 3 Read the GIF image into a BufferedImage

```
BufferedImage image = JDeli.read(gifFile);
```

How to write an image as a GIF file with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.gif"));
```

Step 3 Pass image, Heic type, and File (or OutputStream) object into write method

```
JDeli.write(bufferedImage, "gif", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.GIF, file);
```

or pass in a object for more control over GIF image output, such as level of Compression.

```
GifEncoderOptions options = new GifEncoderOptions();
JDeli.write(bufferedImage, options, file);
```

HEIC files:

What is HEIC?

HEIC stands for “High-Efficiency Image File Format”. It is a complex lossy file format for storing bitmapped images. It was developed by Apple but is an Open Standard.

The file name extension for HEIC files is:

```
.heic
```

How to read a HEIC image in Java with JDeli

Step 1: Add JDeli to your class or module path.
Step 2: Create a File or InputStream pointing to the raw HEIC image. You can also use a byte[] cotaining the image data.
Step 3: Read the HEIC image into a BufferedImage

```
BufferedImage image = JDeli.read(heicFile);
```

How to write an image as a Heic file with JDeli

Step 1: Add JDeli to your class or module path.
Step 2: Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.heic"));
```

Step 3: Pass image, Heic type, and File (or OutputStream) object into the write method

```
JDeli.write(bufferedImage, "heic", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.HEIC, file);
```

or pass in an object for more control over Heic image output, such as level of Compression.

```
HeicEncoderOptions options = new HeicEncoderOptions();
JDeli.write(bufferedImage, options, file);
```

JPEG 2000 files:

What is JPEG2000?

JPEG2000 was developed in 1997-2000 as a replacement for the original JPEG file format. It is not backward compatible. The aim was to produce smaller files than possible with JPEG format. It also uses lossy or lossless compression for digital images.

JPEG2000 is an image decoding standard that was originally created as an update of the well-known JPEG format which was introduced in the year 2000 by the "Joint Photographic Experts Group" committee. It uses lossy compression image mode that is based on discrete wavelet transformation.

The filename extensions commonly associated with JPEG 2000 are:

```
.jp2, .j2k, and .jpx.
```

How to read a JPEG2000 image in Java with JDeli
Step 1 Add JDeli to your class or module path.
Step 2 Create a File, InputStream pointing to the raw JPEG2000 image. You can also use a byte[] containing the image data.
Step 3 Read the JPEG2000 image into a BufferedImage

```
BufferedImage image = JDeli.read(jpeg2000File);
```

How to write an image as a JPEG2000 file with JDeli

Step 1 Add JDeli to your class or module path.

Step 2 Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.jp2"));
```

Step 3 Pass image, JPEG2000 type and File (or OutputStream) object into write method

```
JDeli.write(bufferedImage, "jp2", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.JPEG2000, file);
```

or pass in a Jpeg2000EncoderOptions object for more control over JPEG2000 image output such as level of Compression.

```
Jpeg2000EncoderOptions options = new Jpeg2000EncoderOptions();
JDeli.write(bufferedImage, options, file);
```

JPEG files:

What is JPEG?

JPEG (Joint Photographic Experts Group) is a very well supported file format and works with all modern web browsers. It does not include an opacity channel so there is no transparency.

JPEG works best for colour images. It is commonly used for digital camera images as it gives you high-quality images. One main feature of JPEG is that it enables you to choose between quality and file size; you can sacrifice quality for a smaller file or vice versa.

The file name extension for JPEG files are:

```
.jpg or .jpeg
```

How to read a JPEG image in Java with JDeli

Step 1 Add JDeli to your class or module path
Step 2 Create a File handle, InputStream pointing to the raw JPEG image. You can also use a byte[] containing the image data.
Step 3 Read the JPEG image into a BufferedImage

```
BufferedImage image = JDeli.read(jpegFileOrInputStreamOrByte);
```

How to write out an image as a JPEG file with JDeli

Step 1 Add JDeli to your class or module path.

Step 2 Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.jpx"));
```

Step 3 Pass image, JPEG type and File (or OutputStream) object into write method

```
JDeli.write(bufferedImage, "JPEG", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.JPEG, file);
```

or pass in a JpegEncoderOptions object for more control over JPEG image output such as level of Compression, choice of jp2 or jpx output, etc.

```
JpegEncoderOptions options = new JpegEncoderOptions();
JDeli.write(bufferedImage, options, file);
```

PNG files:

What is a PNG?

PNG stands for Portable Network Graphics. It is a lossless, bitmap image format popular on the world wide web because it supports transparency in browsers. It was created to remove the need for GIF images in web browsers. Unlike GIF, PNG is an open standard with no patents.

The file name extension for PNG files is:
`.png`

How to read a PNG image in Java with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File handle or InputStream pointing to the raw PNG image. You can also use a byte[] containing the image data.
Step 3 Read the PNG image into a BufferedImage

```
BufferedImage image = JDeli.read(pngFile);
```

How to write out an image as a PNG file with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.png"));
```

Step 3 Pass image, PNG type and File (or OutputStream) object into write method

```
JDeli.write(bufferedImage, "PNG", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.PNG, file);
```

or pass in a PngEncoderOptions object for more control over PNG image output such as level of Compression.

```
PngEncoderOptions options = new PngEncoderOptions();
JDeli.write(bufferedImage, options, file);
```

SGI files:

What is SGI file format?

SGI (Silicon Graphics Image) is a bitmap image file format for storing pixel data. It has 2-channel support (.bw for black and white), 3-channel support (rgb/bwa) and 4-channel support (rgba). The ‘a’ represents alpha.

The file name extension for SGI files is:

```
.sgi
```

How to read an SGI image in Java with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File handle, InputStream pointing to the raw SGI image. You can also use a byte[] containing the image data.
Step 3 Read the SGI image into a BufferedImage

```
BufferedImage image = JDeli.read(sgiFile);
```

TIFF files:

What is TIFF?

TIFF stands for “Tag Image File Format”. It is a complex file format for storing one or more bitmapped images in multiple colourspaces using several different forms of compression.

The file name extension for TIFF files is:

```
.tif, .tiff
```

If you are just looking for a TIFF Viewer, JDeli includes a built-in Image Viewer.

How to read a TIFF image in Java with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File handle, InputStream pointing to the raw TIFF image. You can also use a byte[] containing the image data.
Step 3 Read the TIFF image into a BufferedImage

```
BufferedImage image = JDeli.read(tiffFile);
```

How to read a multi-file TIFF image in Java with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File handle, InputStream pointing to the raw TIFF image. You can also use a byte[] containing the image data.
Step 3 Read the TIFF file into a TiffDecoder instance

```
RandomAccessFile raf = new RandomAccessFile("yourTiffFileLocation", "r");
TiffDecoder decoder = new TiffDecoder(raf);
```

Step 4 Loop through the separate images in the TIFF file for

```
(int i = 1; i <= decoder.getPageCount(); i++) {
BufferedImage decodedImage = decoder.read(i);
 // Insert BufferedImage handling code here
}
raf.close();
```

How to write an image as a TIFF file with JDeli

Step 1 Add JDeli to your class or module path.

Step 2 Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.tif"));
```

Step 3 Pass image, TIFF type, and File (or OutputStream) object into write method

```
JDeli.write(bufferedImage, "tiff", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.TIFF, file);
```

or pass in a TiffEncoderOptions object for more control over TIFF image output, such as level of Compression.

```
TiffEncoderOptions options = new TiffEncoderOptions();
JDeli.write(bufferedImage, options, file);
```

WEBP files:

What is WebP?

WebP is a relatively new file format for bitmap images created by Google to work on web browsers. WebP images are designed to be small and therefore fast to load, using both lossy and lossless compression.

The filename extension for WebP files is:

```
.webp
```

How to read a WebP image in Java with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File handle, InputStream pointing to the raw WebP image. You can also use a byte[] containing the image data.
Step 3 Read the WebP image into a BufferedImage

```
BufferedImage image = JDeli.read(webpFile);
```

How to write an image as a WebP file with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File (or OutputStream) object

```
File file = new File("/path/to/outputFile.webp"));
```

Step 3 Pass image, WEBP type, and File (or OutputStream) object into write method

```
JDeli.write(bufferedImage, "webp", file);
```

In JDeli you can also use a typesafe version

```
JDeli.write(bufferedImage, OutputFormat.WEBP, file);
```

or pass in a WebpEncoderOptions object for more control over WebP image output, such as level of Compression.

```
WebpEncoderOptions options = new WebpEncoderOptions();
JDeli.write(bufferedImage, options, file);
```

WMF files:

What is WMF?

Windows Metafile (WMF for short) is an image file format developed by Microsoft, which has the capabilities for both vector and bitmap images. It is more commonly used for its vector capabilities, similar to a file format such as SVG. Note that once you have loaded a WMF file into a BufferedImage, it has been converted into a bitmap image and is no longer a vector format.

The file name extension for WMF files is:

```
.wmf
```

How to read a WMF image in Java with JDeli

Step 1 Add JDeli to your class or module path.
Step 2 Create a File handle, InputStream pointing to the raw WMF image. You can also use a byte[] containing the image data
Step 3 Read the WMF image into a BufferedImage

```
BufferedImage image = JDeli.read(wmfFile);
```

IDR SOLUTIONS

HOW TO CONTACT US

We would love to hear back from you about any suggestions or thoughts you may have

JDELI

Get in touch

support@idrsolutions.com
sales@idrsolutions.com

Zoom Call

Discord

50 Churchill Square,
Kings Hill,
West Malling,
ME19 4YU